(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
5 April 2001 (05.04.2001)

PCT

(10) International Publication Number
WO 01/24474 A1

(51) International Patent Classification[7]: H04L 29/06

(21) International Application Number: PCT/EP00/09204

(22) International Filing Date:
19 September 2000 (19.09.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/156,468 27 September 1999 (27.09.1999) US
09/433,257 4 November 1999 (04.11.1999) US

(71) Applicant: KONINKLIJKE PHILIPS ELECTRONICS N.V. [NL/NL]; Groenewoudseweg 1, NL-5621 BA Eindhoven (NL).

(72) Inventor: SHTEYN, Eugene; Prof. Holstlaan 6, NL-5656 AA Eindhoven (NL).

(74) Agent: WHITE, Andrew, G.; Internationaal Octrooibureau B.V., Prof. Holstlaan 6, NL-5656 AA Eindhoven (NL).

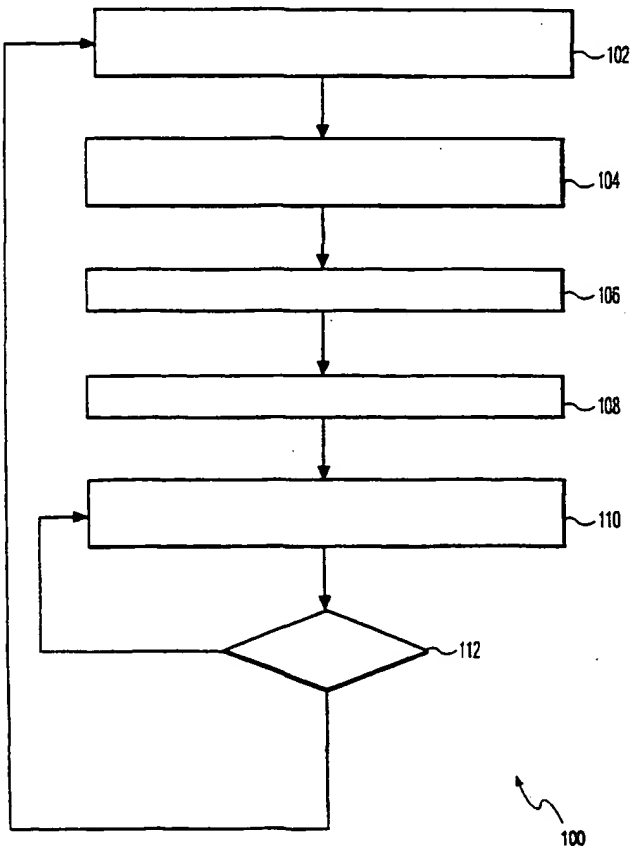
(81) Designated States *(national)*: CN, JP.

(84) Designated States *(regional)*: European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).

Published:
— *With international search report.*
— *Before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments.*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: PARTITIONING OF FILE FOR EMULATING STREAMING



(57) Abstract: An electronic file, e.g., an MP3 file, is partitioned into a sequence of segments at the server side. The first segment is played out (108) upon downloading (106). While the first segment is being played out, the second is being downloaded and buffered (110) so that it is available when the play out of the first segment is completed. While playing out a current one of the segments, next one(s) of the segments are being downloaded and buffered. This partitioning and sequential play out enables to emulate streaming of a file and to minimize latency while downloading an electronic file.

WO 01/24474 A1

Partitioning of file for emulating streaming.

FIELD OF THE INVENTION

The invention relates to content and/or control communications between multiple computer systems, or to such communications between computer systems and consumer devices. Specifically, the invention relates to communication constrained by bandwidth or limited by data processing resources available to the receiving system or device, especially if the communications are received by the user in real time. The type of communications can be, e.g., broadcast, multi-cast or point-to-point.

BACKGROUND ART

Consider current major technologies for delivering digital content, such as audio, video, etc. The streaming method for audio, e.g., RealAudio by RealNetworks, consists of playing-out audio at a client device, while constantly sending data from the server to the client. The technology provided by RealNetworks comprises an encoder, a server, a splitter/cache and a player system with two-way intelligence to resolve network congestion, lost packet conditions and negotiate complex internet protocols. More specifically, the known technology comprises an automatic, variable bit-rate encoding and delivery system for audio and video. The system scales to megabit connection rates and dynamically adjusts the transmission rate as delivery rate varies due to network congestion. The format and the encoding/decoding methods of the data are proprietary. The server and the client synchronize receiving and playing in a way pre-defined by the particular architecture. The communication stack software is tightly coupled to the interpretation layer (application and user interface (UI)). Manufacturers of such technology promote high level of integration between client and server software, as a complete vertical solution. This approach mostly excludes third parties from developing custom server software (e.g., advertizing, services) and/or client applications (UI, special effects, etc.).

Another known method is downloading of a content file from a remote computer with subsequent play-out on the client. MP3 is a widely known audio data format used within the downloading context. There are other data formats, e.g., MP4 for video data etc. The major advantage of the above mentioned method is its open data standard approach.

As long as the right format of the content file is observed during encoding, client and server software/hardware manufacturers are free to develop their own solutions/products.

A major problem with the complete download approach is the inherent latency: there is a delay between the beginning of the download and the start of the play-out. The larger the file and/or smaller the communication bandwidth, the longer it takes to transfer the content from the server to the client. This is particularly undesirable in consumer electronics systems, where perceived delay is detrimental to market acceptance of an open architecture.

SUMMARY OF THE INVENTION

It is an object of the invention to provide an open architecture solution for content delivery in a download approach that allows for a low or negligible play-out latency. To this end the content file is split into multiple parts. Each part or segment requires a relatively short download time. Therefore, the play-out latency is determined by the download time of the first part. The size of the individual part can be determined by the communications bandwidth, e.g., through pinging for a latency-check. The client device/application receives control information about the content. This control information comprises, for example, information relating to the size and memory location of the whole file as well as of it's parts at the server. If the client is not capable of processing split data, it proceeds with the traditional approach, i.e., downloads the whole file and then plays it out. In case the client is capable of processing parts of the content, it uses the relevant control information about the parts in order to continue downloading data, while playing. Data play-out, also called "rendering", is computation-intensive, since it requires a plurality of decoding operations. Data download is bandwidth-intensive. Accordingly, simultaneous play-out and downloading do not significantly compete for the same system resources. This separation between downloading and processing can be efficiently used in a multi-process and/or multi-thread environment.

Preferably, the information contains references to the file location as well as references to the locations of the parts. The intended bandwidth information is associated with the parts. The client may make its own decisions regarding how many parts to download before the start of the play out (execution).

The parts can have different data formats. The format of some of the parts can be proprietary. Information about alternative content parts, regarding bandwidth, format, location access options, etc., can be provided. Content parts can physically reside on different

servers. Content can be split into parts consistent within the semantics of the content, for example, the end of musical phrase, paragraph, target control device, etc. A third party may insert its own content parts in between the original content parts. The third party parts contain, for example, advertisements, commentary, customization options. The format of parts for play-out may be chosen according to user-related information, for example, personal preferences, level of access to premium services, quality of the equipment, bandwidth sharing/fluctuation conditions, etc.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention is explained in further detail and by way of example with reference to the accompanying drawings wherein:

Fig.1 is a flow diagram illustrating the various steps in a method according to the invention; and

Fig.2 gives an example of control code.

Throughout the Figures, same reference labels indicate similar or corresponding features.

PREFERRED EMBODIMENTS

The invention enables emulating the streaming of files while using a download approach. Fig.1 illustrates a flow diagram 100 with various steps involved in the playing-out of a segmented file at the client.

In step 102, the client contacts the server, selects the particular content file and downloads the control information that enables the retrieving and playing out of the segmented file. The control information describes the locations, (URL's), and size of the various file segments, and provides, for example, UI functionalities at the client. In this example, the control information is coded in Extensible Markup Language (XML).

In step 104 the XML code is parsed. Parsing of XML is well known in the art. A person skilled in the art can download an XML interpreter, including source code, from the Internet, see for example the URL www.ibm.com/xml. Thus, the client is enabled to get information about the content information and the URLs of the first and subsequent file segments.

In step 106, the first file segment is downloaded for play-out. Communicating with a remote server is a well known technology. For example, Java 2.0 provides a set of

standard classes that enable retrieving a remote file into a buffer or as a stream. (see, e.g., www.sun.com - java 2.0 java.io.* package documentation).

In step 108, the rendering of the first segment is started. The buffered content of the first segment is forwarded to a decoding/playing module. The decoding/playing module decodes the file format which may be, for example, MP3. The playing of the supplied stream of bits involves a number of standard operating system calls to its drivers-a technique well known in the art.

In step 110, the next file segment is downloaded at the client and stored in a buffer while the previous file segment, here the first file segment, is being played out. One option is to have the downloaded files buffered in a sequence or linked list of buffers. This functionality is typically provided by the operating system of the client. For example, the Microsoft Windows family of products creates a memory buffer associated with the file every time an Application Program Interface (API) call opens the file. Alternatively, in a thread- and/or process-rich environment, several threads and/or processes can be organized to independently retrieve file segments, while playing out the content of other segments. Working with threads is a skill common for software engineers. For example, Java 2.0 from Sun Microsystems provides classes supporting multiple threads (see, for example, java.lang.thread and related documentation). Similarly, Microsoft Software Development Kit (SDK) for the Windows family of products makes thread- or process-related functionalities available to programmers.

Upon completion of playing out the first segment, the second segment is passed on from the buffer to the decoding/playing module. This can be implemented by means of, for example, a linked list. As known, a linked list is a data structure wherein each element (here: segment) has content data and a pointer to a next element (here: next segment).

The decoding/playing module has to decode the file format. The decoding program represents a standard task to a person skilled in the art to program a decoding procedure according to a widely published standard (for example MP3, etc.). The playing of the supplied stream of bits involves a number of standard operating system calls to its drivers - a technique well known in the art.

Fig.2 gives an example of information-describing content coded in XML. The code fragment labels the segments as having a title "The best ever music" performed by "V.R. Famous" and having several parts. The segment labeled "part1" is in a preferred format and described the length of the part, for example, in bytes, the format, the minimum

bandwidth required for a connection, and the location on the Internet. An alternative first part is labeled "part1 alt" having a different length, different format, different minimum bandwidth requirement, and a different location. The XML code can be combined with Extensible Style Language (XSL) for generating a user level UI at the user's client. The client thus can automatically choose the format compatible with the client's play-out capabilities.

When the client has selected the proper file, either the one of which the first part is represented here as in the preferred format or the one in the alternative format, the content of the first part is downloaded from the location specified and playing out is started automatically under application control. Combining multiple sequenced inputs is well understood in the industry. For example, Java Development Kit (JDK) v.1.2 from Sun Microsystems, Inc. provides a class java.io.SequenceInputStream (see http://java.sun.com/products/jdk/1.2/docs/api/java/io/SequenceInputStreams.html) as a standard component of the io class library. SequenceInputStream represents the logical concatenation of other input streams. It starts out with an ordered collection of input streams and reads from the first one until end of file is reached, whereupon it reads from the second one, and so on, until end of file is reached on the last of the contained input streams. An object of the java.io.SequenceInputStream class can be initialized by, e.g., enumeration (see http://java.sun.com/products/jdk/1.2/docs/api/java/util/Enumeration.html) of objects of the InputStream class (see http://java.sun.com/products/jdk/1.2/docs/api/java/io/InputStream.html). This abstract class is the superclass of all classes representing an input stream of bytes. including the class FileInputStream (see http://java.sun.com/products/jdk/1.2/docs/api/java/io FileInputStream). In case of the downloading of multiple file parts, an application can create instances of the FileInputStream class from local temporary files into which the parts are being downloaded. The contents of those multiple local files will be supplied to the Sequencer. The rendering component of the application will read the information out of it as if it were just a single local file.

The segmentation of the content file into separately downloadable segments enables a third party, such as a service provider, to insert between two segments specific content information, e.g., advertisements to be rendered on the client's display.

During operation, the client application could select a next segment in a different format for the same content to adapt to changing circumstances, e.g., lower bandwidth due to network congestion. Also, the user could be prompted to subscribe to a

service that as a demo lets the user download only the first segment in a high quality and the next segments in a lower quality. The combination of XML and a corresponding parser and interpreter at the client controls the downloading and playing out as explained above. Accordingly, the client pulls the content segments from the locations indicated in the XML control information for buffering and subsequent play-out.

The implementation of a client in this client-server architecture can be done in a variety of ways. A first example is a hardware-based single-purpose device, similar to the Rio MP3 player by the Diamond Corp. In order to accommodate the method of the invention, the player needs, in addition, an XML parser, the ability to interpret XML and the ability to download and play-out content segments sequentially. A second example is to implement the method of the invention as a software application on a multi-purpose computing device, e.g., a PC or a set top box. The device has the software implementing the functionalities mentioned above. In a graphics-rich environment, multiple GUI's are represented to the user for further customization.

The following co-pending applications are incorporated herein by reference:

U.S. serial no. (Attorney docket PHA 23,768) filed 9/27/99 for Raoul Mallart for SCALABLE SYSTEM FOR VIDEO-ON-DEMAND. This patent document relates to a Video-on-Demand service (VOD) that is emulated in a Near-Video-on-Demand (NVOD) architecture. Content information is made available to an end-user in the NVOD architecture. An introductory portion of the content information is stored at the end-user's equipment, e.g., by downloading overnight. During playing out of the introductory portion at the end-user enabling the content information supplied in the NVOD architecture is buffered at the end-user's equipment. The equipment is controlled to switch from playing out the introductory portion stored to playing out the buffered content information.

U.S. serial no. 09/189,534 (Attorney docket PHA 23,528) filed 11/10/98 for Eugene Shteyn for CONTENT SUPPLIED AS SOFTWARE OBJECTS FOR COPYRIGHT PROTECTION. This document relates to supplying content information such as a movie, an audio file or a textual message to an end-user in a software object. The object has an encapsulated procedure for end-user access of the content information in a runtime environment. The object can specify time frame for and manner wherein the content information is to be accessed. Since the procedure is encapsulated in the object together with the content data, and since transport of the object over the Internet is done after serializing, an adequate degree of security is provided against unauthorized play-out or copying.

U.S. serial no.09/149,950 (Attorney docket PHA 23,495) filed 9/9/98 for Raoul Mallart for REAL TIME VIDEO GAME USES EMULATION OF STREAMING OVER THE INTERNET IN A BROADCAST EVENT. This patent document relates to emulating streaming of animation data over the Internet to a large number of clients in a broadcast application on a client-server network. The animation is considered a sequence of states. State information is sent to the clients instead of the graphics data itself. The clients generate the animation data itself under control of the state information. The server and clients communicate using a shared object protocol. Thus, streaming is accomplished as well as a broadcast without running into severe network bandwidth problems. This approach is used to map a real life event, e.g., a motor race, onto a virtual environment in order to let the user participate in a virtual race against the real life professionals, the dynamics of the virtual environment being determined by the state changes sent to the user.

U.S. serial no. 09/138,782 (Attorney docket PHA 23,491) filed 8/24/98 for Raoul Mallart and Atul Sinha for EMULATION OF STREAMING OVER THE INTERNET IN A BROADCAST APPLICATION. In a broadcast application on a client-server network the streaming is emulated of animation data over the Internet to a large number of clients. The animation is considered a sequence of states. State information is sent to the clients instead of the graphics data itself. The clients generate the animation data itself under control of the state information. The server and clients communicate using a shared object protocol. Thus, streaming is accomplished as well as a broadcast without running into severe network bandwidth problems.

U.S. serial no. 09/283,545 (attorney docket PHA 23,633) filed 4/1/99 for Eugene Shteyn for TIME- AND LOCATION-DRIVEN PERSONALIZED TV. This document relates to a service for personalized video recorders such as the one from TiVo-Philips. The recorder has a hard-disk that serves as a random-access buffer.

From reading the present disclosure, the other modifications will be apparent to persons skilled in the art. Such modifications may involve other features which are already known in the design, manufacture and use of systems and devices and component parts thereof and which may be used instead of or in addition to features already described herein.

CLAIMS:

1. A method of enabling to emulate streaming of a file over a data network to a client, the method comprising:

partitioning the file into multiple segments;

enabling the client to download (106) a first one of the segments for playing out;

enabling the client to download (110) a next one of the segments while playing out a current one of the segments;

enabling the client to buffer the next segment while playing out the current segment; and

enabling the client to start playing out the buffered next segment upon completion of the playing out of the current segment.

2. The method of claim 1, wherein the partitioning is determined by information about the client.

3. The method of claim 1 or 2, wherein the partitioning is determined by information about the network.

4. The method of claim 1,2 or3, wherein the file comprises an audio file.

5. The method of claim 1,2 or 3, wherein the file comprises a video file.

6. The method of anyone or more of claims 1 to 5, wherein the partitioning comprises adding respective tags to respective ones of the segments.

7. An electronic file comprising information content partitioned into multiple segments that are separately downloadable over a data network, the file comprising control information for enabling to play out a first one of the segments upon downloading, enabling

to buffer a second one of the segments while the first segment is being played out and enabling a seamless transition between the playing out of the first and the second segments.

8. The file of claim 7, wherein a respective one of the multiple segments comprises respective control information.

9. The file of claim 7 or 8, implemented as a linked list.

10. The file of claim 7, 8 or 9, comprising the control information in Extensible Markup Language (XML) format.

11. A device for play-out of information content received over a data network from a server, wherein:

- the information content comprises multiple segments;
- the device is capable of downloading a first one of the segments from the server for playing out;
- the device is capable of downloading a next one of the segments while playing out a current one of the segments;
- the device is capable of buffering the next segment while playing out the current segment; and
- the device is capable of starting to play out the buffered next segment upon completion of the playing out of the current segment.

12. The device of claim 11, wherein:

- the content information is accessible through control information provided to the device; and
- the device is capable of interpreting the control information to retrieve the segments from the server for sequential play-out.

13. The device of claim 12, wherein:

- the control information comprises an XML format;
- the device has an XML parser; and
- the device has an XML interpreter.

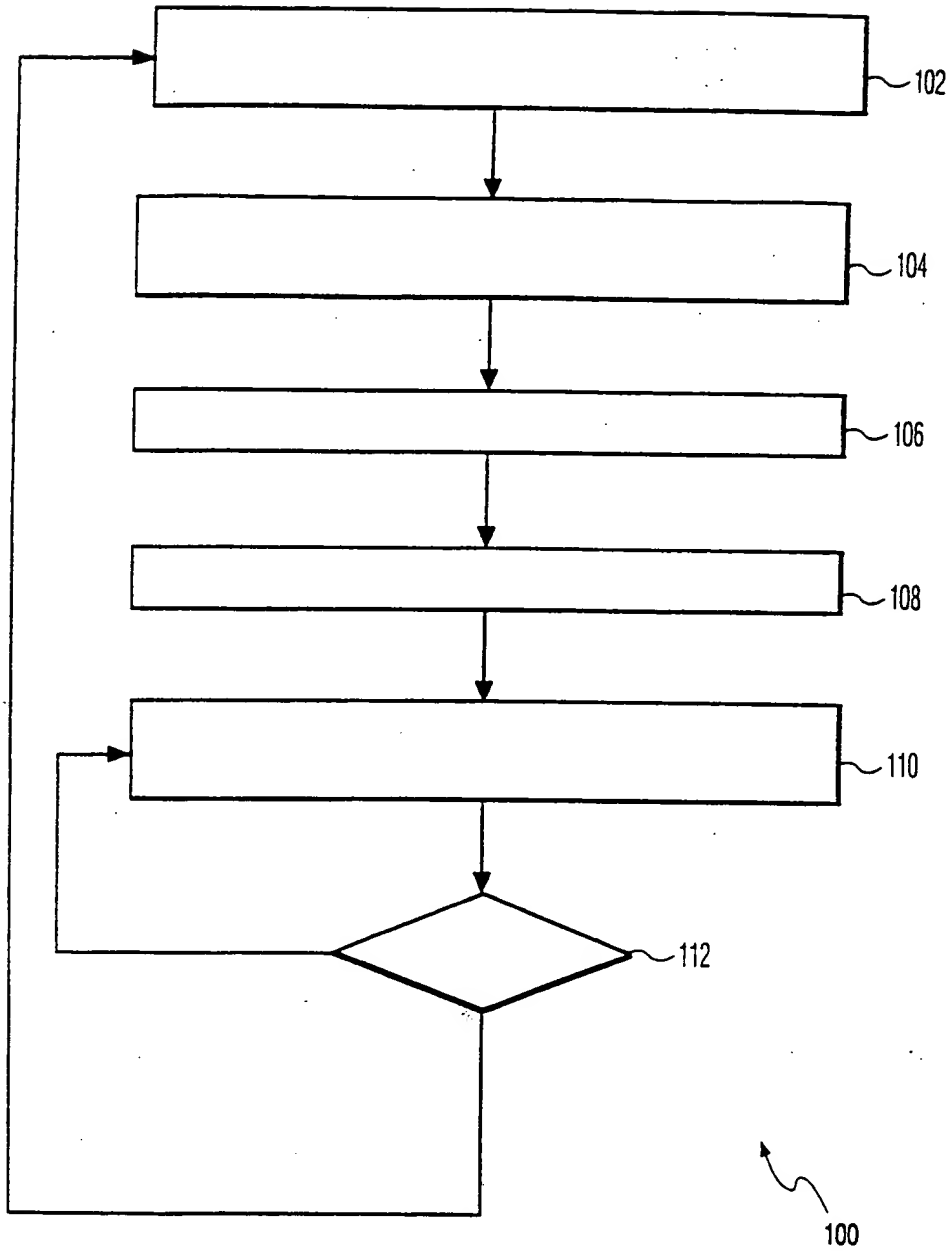


FIG. 1

```
<XML>
    <title>
    The best ever music
    </title>
    <artist>
    V.R. Famous
    </artist>
    <parts>                    (Preferred format)
        <part1>
            <length> 1024 </length>
            <format> MP3 </format>
            <location> ftp://137.27.52.87 </location>
            <min_bandwidth> 10,000 </min_bandwidth>
        </part1>

                               (Alternative format)
        </part1_alt>
            <length> 512 </length>
            <format> OTHER </format>
            <location> http:// yevgeniynet/ .... _</location>
            <min_bandwidth> 8,000 </min_bandwidth>
        </part_alt1>

.....

    </parts>
</XML>
```

FIG. 2

INTERNATIONAL SEARCH REPORT

Int[illegible]nal Application No
PCT/EP 00/09204

A. CLASSIFICATION OF SUBJECT MATTER
IPC 7 H04L29/06

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC 7 H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal, WPI Data, PAJ, INSPEC, IBM-TDB

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X Y	US 5 751 968 A (COHEN ALON) 12 May 1998 (1998-05-12) column 4, line 28 -column 5, line 4; figure 1 column 5, line 24 - line 53; figure 2 column 6, line 6 - line 61; figures 4,5	1,4-8, 11,12 10,13
X A	US 5 442 390 A (GOLDMAN MATTHEW S ET AL) 15 August 1995 (1995-08-15) column 8, line 43 -column 9, line 18; figures 1-3,5 column 9, line 64 -column 10, line 36; figure 6	1,6-8, 11,12 2,3

-/--

[X] Further documents are listed in the continuation of box C.

[X] Patent family members are listed in annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
16 February 2001	02/03/2001

Name and mailing address of the ISA	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 epo nl, Fax: (+31-70) 340-3016	Polzer, A

Form PCT/ISA/210 (second sheet) (July 1992)

C.(Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT

Category °	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	EP 0 923 033 A (TOYOTA MOTOR CO LTD) 16 June 1999 (1999-06-16) paragraph '0021! - paragraph '0024!; figures 2,3	1,7,11
P,Y	GIRARDOT M ET AL: "Efficient representation and streaming of XML content over the Internet medium" 2000 IEEE INTERNATIONAL CONFERENCE ON MULTIMEDIA AND EXPO. ICME2000. PROCEEDINGS. LATEST ADVANCES IN THE FAST CHANGING WORLD OF MULTIMEDIA (CAT. NO.00TH8532), PROCEEDINGS OF INTERNATIONAL CONFERENCE ON MULTIMEDIA AND EXPO, NEW YORK, NY, USA, vol. 1, 30 July 2000 (2000-07-30) - 2 August 2000 (2000-08-02), pages 67-70, XP002160596 Piscataway, NJ, USA ISBN: 0-7803-6536-4 section 3.2; figure 2	10,13

1

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No
PCT/EP 00/09204

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5751968 A	12-05-1998	IL 115263 A	11-04-1999
		AU 712229 B	04-11-1999
		AU 6888796 A	17-04-1997
		CA 2230259 A	03-04-1997
		EP 0850451 A	01-07-1998
		WO 9712447 A	03-04-1997
		JP 11512893 T	02-11-1999
US 5442390 A	15-08-1995	CA 2127347 A	08-01-1995
		DE 69415880 D	25-02-1999
		DE 69415880 T	19-08-1999
		EP 0633694 A	11-01-1995
EP 0923033 A	16-06-1999	JP 10069450 A	10-03-1998
		WO 9809222 A	05-03-1998